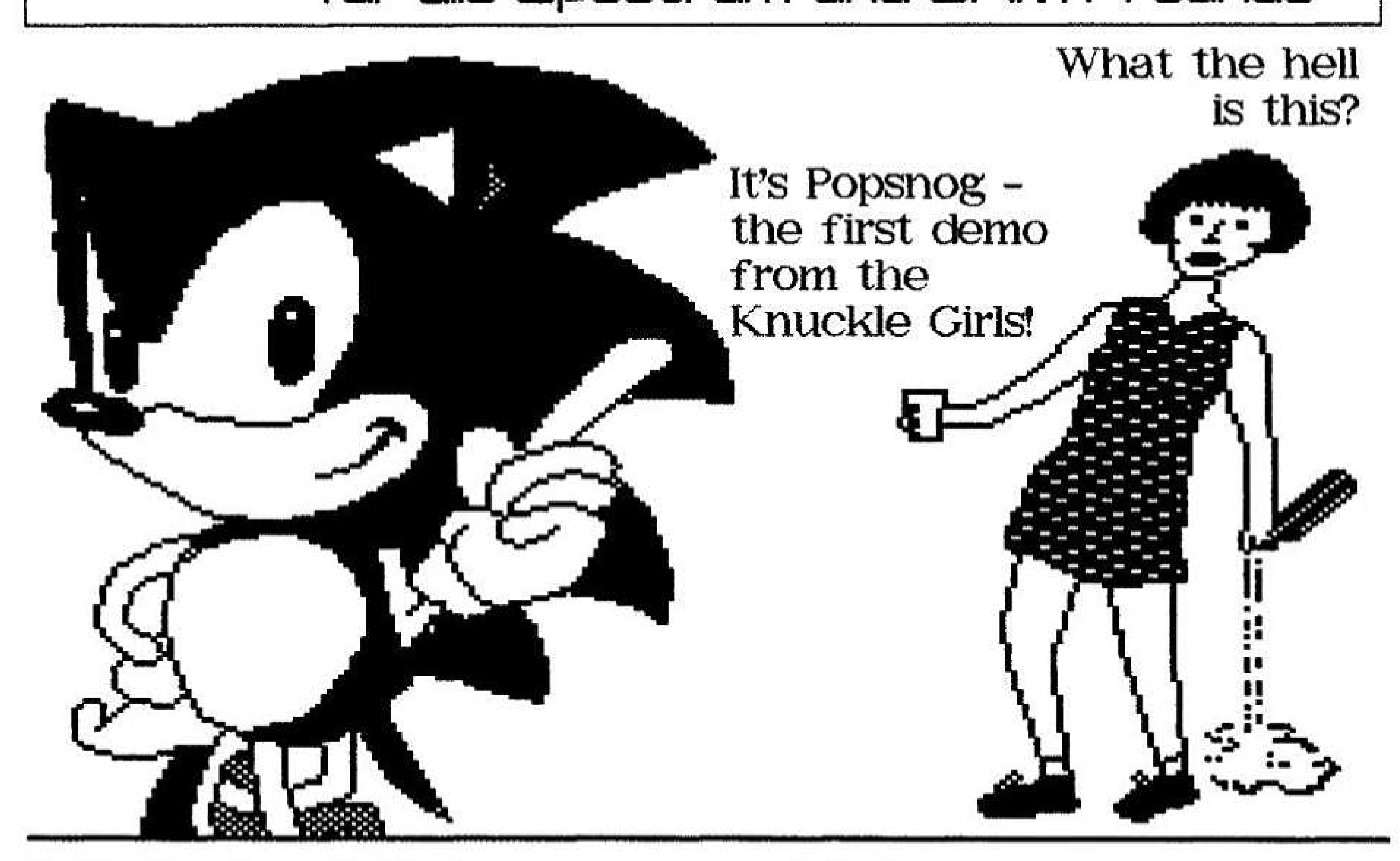
Spectrum Profi Club

für alle Spectrum und SAM Freunde



Wolfgang und Monika Haller
Ernastr. 33, 51069 Köln, Tel. 0221/685946
Bankverbindung: Dellbrücker Volksbank
BLZ 370 604 26, Konto-Nr. 7404 172 012

INFO

Sept. 1994

Smalltalk

Hurra!

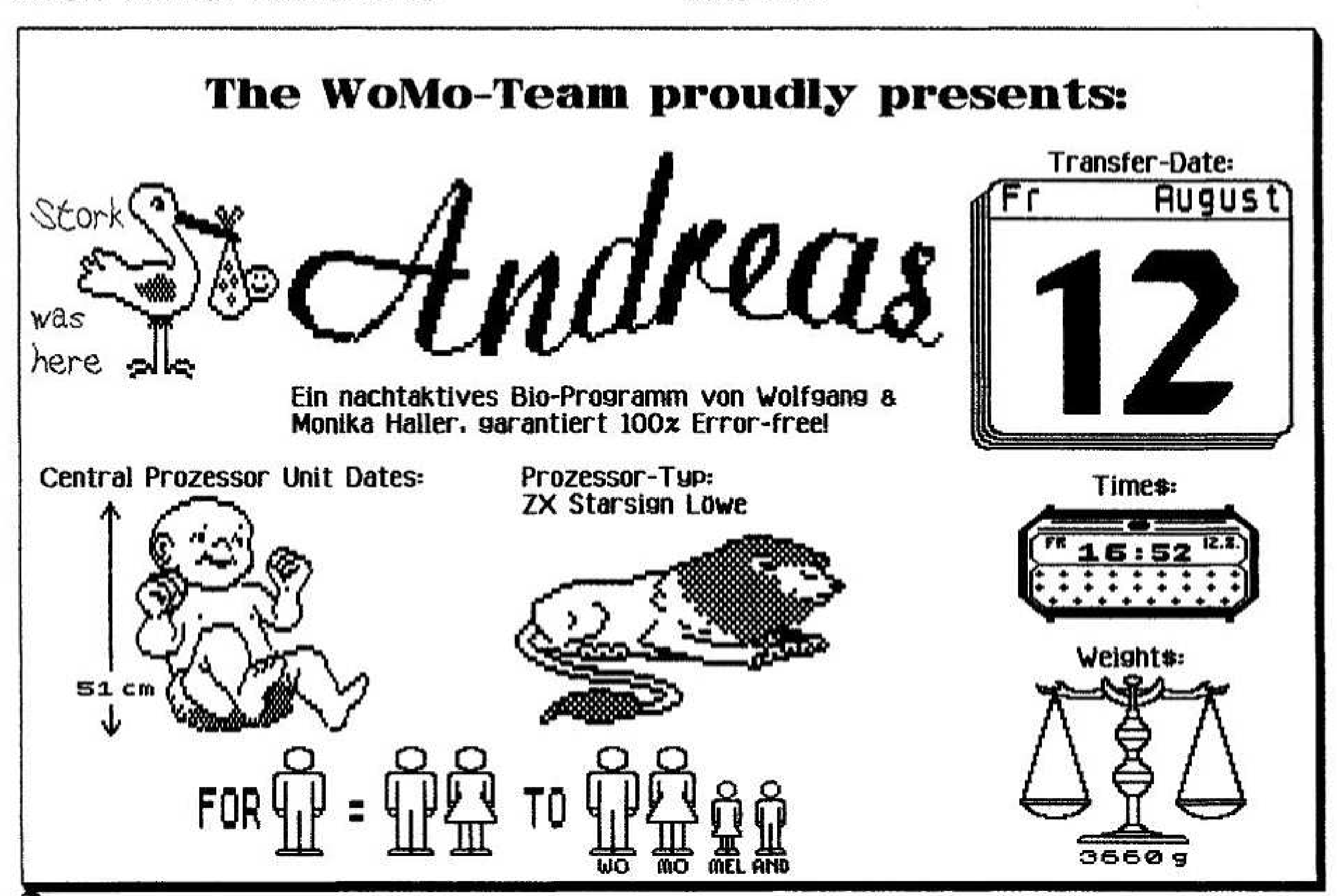
Wir haben erneut Nachwuchs bekommen und uns erlaubt, dies mit untenstehender Anzeige euren "ungläubigen" Augen kundzutun. Somit ist auch klar erkenntlich, das wir nicht nur am Spectrum oder SAM hängen, sondern auch sonst ganz beachtliche Erfolge erzielen können.
Dies bedeutet aber auch, das Moni sich fast ganz aus dem aktiven Clubleben zugunsten der Kinder zurückziehen wird, und fast alle Clubaktivitäten von mir alleine erledigt werden. Dennoch bleiben wir euch als WoMo-Team erhalten, nicht zuletzt deshalb, weil es mir ohne Monikas Unterstützung, Ermunterung und Billigung nicht möglich wäre, soviel Zeitaufwand für den SPC aufzubringen. Selbstverständlich steht euch Monika aber immer noch telefonisch mit Rat zur Seite.
Alle, die noch auf Post von uns warten, bitte ich um ein wenig Geduld, bis sich hier alles etwas besser eingespielt hat. Und so ganz allmählich kommen wir auch wieder "in den Tritt".
Für eure Glückwünsche sind wir natürlich jederzeit empfänglich. Tut euch also keinen Zwang an, ebenso nicht mit Artikeln, Tips, Tricks usw. fürs Info.
Noch eine ganz wichtige Bitte: Ruft uns nicht mehr nach 21 Uhr an. WIR BRAUCHEN UNSEREN SCHLAF ZUR ZEIT MEHR DENN JE!
Für euer Verständnis bedanken wir uns im voraus. Doch nicht nur wir haben Zuwachs bekommen, auch der SPC hat gleich drei neue Mitglieder bekommen, sodaß der aktuelle Mitgliederstand 151 User beträgt. Es sind dies:

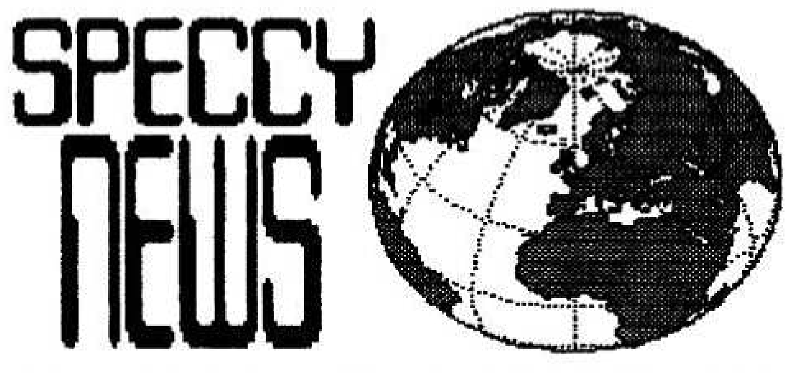
Ed Polinski, Boczna 27
PL-05-300 Minsk Maz, Polen
Wolfgang Rapp, Säntisstraße 4
88145 Hergatz und
Preben D. Sörensen, Jaegersgardgade 128
DK-8000 Arhus C., Dänemark

Auch eine Adressänderung gibt es zu vermelden:

Ludwig Halbritter, Wohlleibstraße 16
86459 Gessertshausen

Wir hoffen, daß dieses unter "erschwerten" Umständen entstandene Info auch diesmal wieder für jeden etwas bietet.
Für die kommenden Infos bitten wir euch, eure Anzeigen, Artikel usw. bis jeweils zum 25. eines Monats spätestens zu schicken. Textfiles, bzw. Screen$ auf Kassette, Opus oder Plus D Diskette sind immer willkommen und die Datenträger schicken wir euch auf Wunsch auch wieder zurück.
Und nun viel Spaß am Info, das (Extended) WoMo-Team



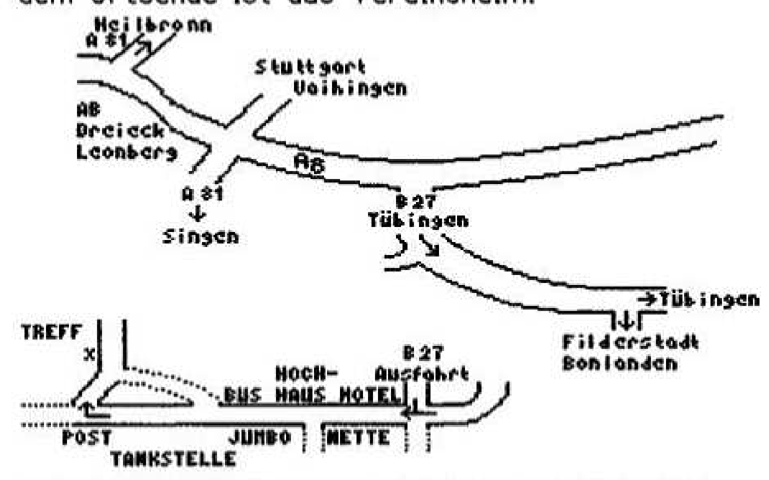


AUF NACH FILDERSTADT

Wir weisen heute nochmal auf das Treffen in Filderstadt hin, welches am

Samstag, dem 8. Oktober 1994 von 10 - 20 Uhr

am gleichen Ort wie bisher stattfindet. Hier noch einmal kurz eine Wegbeschreibung: Von Richtung München oder Karlsruhe auf die A8. Beim Flughafen auf die B 27 Richtung Tübingen / Reutlingen wechseln. Auf der B 27 bleibt man bis zur Ausfahrt **Filderstadt-Bonlanden.** Nach der Ausfahrt Richtung Ortskern fahren und bei der Post rechts abbiegen. Das letzte Gebäude vor dem Ortsende ist das Vereinsheim.



NEUES DISK-INTERFACE AUS DER TSCHECHEI?

Es gibt ja nun schon viele Disk-Interfaces, ein Neues, bisher nur 5-mal hergestelltes und dann vom Erbauer wegen angeblich geringer Nachfrage eingestelltes, soll nun mittels einer Initiative, auf den Markt gebracht werden. Es handelt sich um ein durch einen einfachen Poke aktivierbares IF, welches dann mit den normalen Tapebefehlen arbeitet (wie DEVICE "T" oder DEVICE "D" beim SAM). Dies ist besonders interessant bei Multiloadern oder Headerless-Files. Außerdem verwaltet das IF auf 83 Tracks 1.857.536 Bytes, zur Verwaltung dieser Datenmenge ist der Katalog in 4 Directories unterteilt.

Damit dieses IF wieder zur Herstellung gelangt, sollte man an JSH, Oldrich Palenicek, Sumice 110, 68731 Okr. Uh. Hradiste, Czech Republik schreiben. Dieser sammelt die Schreiben der interessierten User und legt diese dem Erbauer vor, damit dieser die Baupläne freigibt. Wer noch weitere Fragen dazu hat, kann sich mit Thomas Eberle vom SUC in Verbindung setzen.

SOFTWARE SAMMELBESTELLUNG

Von Thomas erfuhren wir, das sich auf die Liste im Info 8/94 nur 1 Mitglied des SPC bei ihm gemeldet und eine Bestellung abgegeben hat. Deshalb dies hier noch einmal zur Erinnerung. Desweiteren kommen zur Liste noch folgende Programme hinzu:

Electra 9000, Excalibur, Gateway to Hell, Ginormous ollection, Kid Pack (59%), Popeye Collection (83%), Postman Pat Collection (50%), Seasurfer, Star Paws (80%), Starburst, Tower of Evil und Wiz Biz.

Ansonsten gelten die gleichen Bedingungen wie im Augustinfo auf Seite 7 aufgeführt. Interessierte wenden sich auch hier bitte an Thomas Eberle, Gastäckerstr. 23, 70794 Filderstadt, Tel. 0771/777142.

ALCHNEWS 13

von Alchemist Research ist draussen und umfangreicher denn je. Dort findet ihr auch ein Interview mit dem WoMo-Team und viele andere interessante Dinge. Wendet euch an: Andy Davis, 62 Tithe Barn Lane, Woodhouse, Sheffield, South Yorkshire, S13 7LN, England (Tape oder Disk + Rückporto) oder an uns (dto.).

BROTKASTEN LIVE

ist ein neues Heft von Thomas Eberle (aus Maulbronn!), welches uns durch Zufall in die Hände fiel. Es beinhaltet überwiegend Seiten für den C64 (Brotkasten), aber auch für alle anderen Systeme, so auch für den Spectrum. Es erscheint jetzt 2-monatlich im Wechsel mit dem CF und kostet 7,80 DM!!! Für uns Clubs zum inserieren sicher interessant, ansonsten bildet euch eine eigene Meinung...

SINCLAIR CLASSIC 7a/b

ist ebenfalls erschienen. Dank Stefan Baller-staller ist der SPC nun als Distributor für Deutschland genannt, somit ist dieses nett gemachte Tape-/Diskmag ganz offiziell gegen Tape oder Disk + Rückporto von uns erhältlich.

SUBLIMINAL EXTACY 1

ist ein neues Tape-/Diskmag von Extacy III. Die erste Ausgabe wurde fast ausschließlich von L.A. 14 Deverose Court, Hanham, Bristol, BS15 3SW und Bogie, 28 Wilshire Avenue, Hanham, Bristol, BS15 3QT (England) gemacht. Es ist PD und kann von uns bezogen werden (Tape oder Disk + Rückporto).

POPSNOG

ist das erste weibliche Demo (!) und ebenfalls durch uns erhältlich. Es enthält einen Screen, einen selbstgemachten ST-Song und kernige Sprüche über die "Boys". Wer diesem "Girl" schreiben will, hier die Adresse: Amy Pandagirl, c/O 2 Belvoir Road, Nottingham, NG2 5DL, England.

DIE SEITEN FÜR DEN SAM!

Tastatur 'Peeken'

Wie alle wissen, können wir in einen Memory Bereich 'peeken', aber es gibt auch die Möglichkeit, die Tastatur zu 'peeken'. OK, es ist kein echtes 'PEEK', es ist jedoch oft sehr von Nutzen, die Tastatur direkt abzulesen, insbesondere in einem Maschinencode Programm, aber auch manchmal in Basic (z.B. um die Cursor-Tasten zu lesen).

Da unser SAM Spectrum kompatibel sein sollte, ist aus diesem Grund auch die Tastatur genau wie die des Spectrum aufgebaut, das heißt, das der PORT 254 für die Tastatur reserviert ist:

```
BASIC: LET tast=(IN xxxx) BAND 31

MC:     LD    BC,xxxx
        IN    A,(C)
        AND   31
```

Der Tastenzustand wird in die unteren 5 Bits reflektiert und die anderen Bits werden durch den AND-Befehl ausgeblendet. Wenn keine Taste in der Gruppe gedrückt ist, bekommt man normalerweise 31 (Hex &1F) als Datenwert.

PORT 254 bedeutet, das die unteren 8 Bits der Portadresse immer Hex &FE sind. Die Tasten sind in Gruppen unterteilt, um jede Gruppe abzulesen setzt man ein einziges Bit in den oberen 8 Bits der Adresse auf Null.

	D4	D3	D2	D1	D0
7FFE (32766)	B	N	M	S/S	SP
BFFE (49150)	H	J	K	L	ENT
DFFE (57342)	Y	U	I	O	P
EFFE (61438)	6	7	8	9	0
F7FE (63486)	5	4	3	2	1
FBFE (64510)	T	R	E	W	Q
FDFE (65022)	G	F	D	S	A
FEFE (65278)	V	C	X	Z	Caps

So ergibt z.B. 'LET tast=IN 32766 BAND 31' bei gedrückter SPACE-Taste für die Variable 'tast' den Wert 30 (Bit D0 auf Null).

Der SAM hat jedoch mehr Tasten als ein Spectrum. Somit stellt sich die Frage, wie wir diese zusätzlichen Tasten lesen können. Dazu benutzt der SAM 3 Bits des PORT 249.

	D7	D6	D5
FEF9 (65273)	F3	F2	F1
FDF9 (65017)	F6	F5	F4
FBF9 (64505)	F9	F8	F7
F7F9 (63481)	CAPS	TAB	ESC
EFF9 (61433)	DEL	+	-
DFF9 (57337)	F0	"	=
BFF9 (49145)	EDIT	:	;
7FF9 (32761)	INV	.	,

Nun gibt es noch weitere 5 Tasten, die 4 Cursortasten und die CNTRL-Taste, welche ebenfalls einfach über den PORT 254 gelesen werden:

	D4	D3	D2	D1	D0
FFFE (65534)	→	←	↓	↑	CNTRL

Natürlich sehen wir durch unseren IN-Befehl nur ein Binär-Bit für jede Taste auf Null gesetzt. Es ist jedoch auch möglich, den ASCII-Wert für jede Taste durch Nutzen einer ROM Routine abzulesen. Diese Routine heißt 'JWAITKEY' und wird folgendermaßen aufgerufen:

```
BASIC:    GET a$
          LET a=CODE a$

MC:       CALL &016C
          (Daten in Register A)
```

Für jede gedrückte Taste bekommt man dann einen ASCII-Wert geliefert. Die 'extra' Tasten des SAM liefern folgende Codes:

F0=192	F1=193	F2=194	F3=195
F4=196	F5=197	F6=198	F7=199
F8=200	F9=201		
DEL=12	EDIT=7	TAB=252	
← =8	→ =9	↓ =10	↑ =11

Viel Spaß beim Tastatur-'Peeken'. Noch ein schneller Tip für alle, die in Maschinencode programmieren, auf FRED 47 ist ein Super MC Debug Programm, welches GI_MON heißt.

Ian D. Spencer, Fichtenweg 10c
53804 Much, Tel. 02245/1657



Was tut sich beim SAM? (Teil 2)

Hallo SAM-Fans!

Heute setzen wir unsere Übersicht über die bestehenden SAM Programme vom letzten Info fort. Außer Ians Artikel auf Seite 4 herrschte von den anderen SAM Usern wieder einmal das große Schweigen.
Nun, durch die Geburt unseres "Söhnchens Andreas" sind wir natürlich auch ein wenig aus dem "Takt geraten", deshalb blieb die Erfassung der Programme aus den SAM Supplements leider auf der Strecke. Eines kann man aber schon generell sagen: Der Inhalt der SAM Supplements ist weniger Spiele- und Demoorientiert, dafür aber mehr technisch, ebenso gibt es eine Reihe von Hilfsfiles, wo z.B. der MEM$-Befehl, der Sound-Chip oder das Cutting mit Flash näher erläutert und gegebenenfalls mit Demos unterlegt werden. Dazu aber vielleicht demnächst mehr.
Heute wollen wir uns also den generellen Programmen und nächstesmal (aus Platzgründen) den Utilities widmen.

Programme

Es ist nicht immer leicht, Programme eindeutig einer Sparte zuzuordnen, deshalb fassen wir sie hier allgemein zusammen. Da gibt es zum Beispiel den "Adress Manager" (Fred 6) von J. N. Coelho oder "Filer" (Fred 7) von Scott Preston, Programme zur Namens- und Adressverwaltung. Oder "Amateur Radio Multimode" von Ian D. Spencer, unserem fleißigsten am SAM, er hatte ja auch eine tolle Serie über das Senden und Empfangen per RTTY, Amtor usw. im Info geschrieben.
Wer sich für "Artifical Intelligence" (Arcadia 4) begeistert, dem sei dieses Programm eines "Neural Network Simulators" empfohlen. Auf gar keinen Fall darf ein Biorhythmus-Programm (Enceladus 12) in einer Sammlung fehlen, dieses hier stammt von Armughan Javaid aus Pakistan! Mit "Chemistry Test" (Fred 35) von Marc Broster kann man seine Chemie- (und englisch) Kenntnisse prüfen.
"Datapost" (Fred 5) von CM registriert die Postein- und -ausgänge (es geht allerdings auch ohne!). "Demon Animator" (Fred 18, Paul Miner), "Life" (Fred 31, N. McLean) und "Life Animator" (Fred 18, Paul Milner) beschäftigen sich alle mit den berühmten Zellen und deren Entwicklung, was aber die Sache am SAM so interessant macht, ist der Einsatz von Farben. Dies gilt ebenfalls für "Mandelbrots" (Fred 20, Paul Milner).
Wer möchte, kann auch per SAM ein Tagebuch führen, "Diary" (Fred 6, Stephen Taylor) hilft dabei. "Dictionary" (Fred 43, Luke Falla) ist ein Vokabeltester, der sogar ein "German"-File enthält. Witzig ist "Dinosaurs" (Fred 44, NoName), mit dem man sich die lustigsten Dinosaurier selber zusammenstellen kann (achtet dann mal auf die Namen). Und da wir beim Zusammenstellen sind: wie wär's mit einem eigenen Universum per "Universe Simulator" (Fred 41, Andrew Collier)?
Erwähnen wir noch "Maths" (Fred 20, Colin Allett oder Turbo 15, P. Mercer), zwei Programme zum Mathe üben und Planet Drawer (Enceladus 12, Steven Ward) bzw. "Solar System Viewer" (Enceladus 5) für die Astronomen unter euch.

Und nun einige generelle Tips...

Um sicherzustellen, daß ein Screen bei 256K oder 512K an die richtige Stelle geladen wird, genügt folgende Zeile, die herausfindet, wo der Screen$ beginnt:

```
10 LET SC=[[252 BAND 31]+1]*16384
```

Listings können wie beim Spectrum auch farbig gehalten werden, um z.B. bestimmte Bereiche zu markieren. Das geht ganz einfach mit:

```
CTRL-I [Nr.] = für Pen [Ink]
CRTL-P [Nr.] = für Paper
```

Probleme mit dem Ramtop lassen sich umgehen, indem man einen Extra-Speicherbereich reserviert, z.B.:

```
IF RAMTOP<81919 THEN OPEN TO 4
CLEAR 81919
```

...sowie Spieletips:

4 Passwörter für PIPEMANIA: Level 7 = Disc, Level 13 = Chip, Level 19 = Mags und Level 31 = Samy.
9 Passwörter für BULGULATORS: Fox, Lie, Ale, Rat, Int, Key, Neb, Gin und Off.
10 Passwörter für HEXAGONIA: Level 2 = Magical, Level 3 = Atomics, Level 4 = Fantasy, Level 5 = Khazad Dum, Level 6 = Disaster, Level 7 = Symphony, Level 8 = X Rays, Level 9 = Level 9, Level 10 = Blah Blah und Level 11 = Dungeon.
Cheat für PARALLAX: soviele Tasten wie möglich im Menu-Screen drücken.

Vielleicht habt ihr ja auch ein paar Tips auf Lager, die ihr weitergeben könnt? Wir würden uns freuen. Bis zum nächsten Info

Das WoMo-Team

Wickeldatenrechner

Für Drehstrommotoren in Sternschaltung

Für PC's kann man dafür Programme kaufen, die teurer sind als unser ganzer Computerkram, und noch recht umständlich dazu. Mein Nachbar hat meinen Betrieb übernommen und ich habe ihm jetzt dieses Programm geschrieben und betont einfach gehalten. Trotzdem kann z.B. ein Reparaturbetrieb oder Kleinbetrieb für elektrische Spezialmaschinen sehr viel damit anfangen. Nun wartet er darauf, ob ihm sein PC-Händler dieses einfache Programm auf seinen PC übertragen kann. Basic kann er angeblich, also, wir werden sehen ob ein PC in der Lage ist, so etwas auch zu machen.

Der Rechner fragt nach Netzspannung, Polzahl des Motors, Frequenz, Durchmesser der Statorbohrung (Rotordurchmesser) in cm, Eisenlänge in cm, Induktion in Gauß und Nutenzahl im Stator.

```
   5 CLS : PRINT ';TAB 6;"WICKELDATEN
     RECHNER"
  10 PRINT #0;AT 0,0;"EINGABE: ": PAUSE
     0: FOR x=1 TO 7: GO SUB 100+10*x:
     NEXT x
  20 PRINT AT 14,0;"OK j/n,aendern mit
     Zahlen 1-7": PAUSE 0: IF INKEY$=
     "j" THEN  GO SUB 200: GO TO 20
  30 LET x=VAL INKEY$: GO SUB 100+10*
     x: GO TO 20
 100 INPUT "Nutquerschnitt: ";q$: IF
     q$="" THEN  INPUT "Drahtstaerke:
      ";s;"Drahtzahl pro Nute: ";z3:
     LET q=PI*s*s*z3/4: GO TO 104
 102 LET q=VAL q$*.28
 104 PRINT AT 12,3;"0) Drahtstaerke
      ";SQR (((q/z)*4)/PI);" mm":
     RETURN
 110 INPUT "Netzspannung: ";u: PRINT
     AT 3,3;"1) Netzspannung: ";u;" V
     ": RETURN
 120 INPUT "Polzahl :";p: PRINT AT 4,
     3;"2) Polzahl: ";p: RETURN
 130 INPUT "Frequenz: ";f$: IF f$=""
     THEN  LET f$="50"
 132 LET f=VAL f$: PRINT AT 5,3;"3)
     Frequenz: ";f;" Hz": RETURN
 140 INPUT "Durchmesser: ";d: PRINT AT
     6,3;"4) Durchmesser: ";d;" cm":
     RETURN
 150 INPUT "Laenge: ";l: PRINT AT 7,3;
     "5) Laenge: ";l;" cm": RETURN
 160 INPUT "Luftinduktion: ";b: PRINT
     AT 8,3;"6) Induktion: ";b;" Gauss"
     : RETURN
 170 INPUT "Nutenzahl: ";a: PRINT AT
     9,3;"7) Nutenzahl: ";a
 172 PRINT AT 10,3;"8) Induktion: 4000
     -8000 Gauss";AT 11,3;"9) Leistungs
     schaetzung";AT 12,3;"0) Draht
     staerke ermitteln": RETURN
 180 FOR b=4000 TO 8000 STEP 100:
     PRINT AT 16,6;"Induktion: ";b;
     " Gauss": GO SUB 200: PAUSE 0: IF
     INKEY$="a" THEN  GO TO 20
 182 NEXT b: RETURN
 190 INPUT "Konstante C: ";c: PRINT AT
     18,6;"Leistung: ";: LET n=c*d*d*l*
     60*f*20e-4/p: PRINT n;" W";TAB 16;
     n/736;" PS": INPUT "Nochmal?:";j$:
     IF j$="j" THEN  GO TO 190
 192 RETURN
 200 LET z1=0.788*u*p*1e8: LET z2=2*f*d
     *l*b*a: LET z=z1/z2: PRINT AT 15,
     6;"Draehte pro Nute: ";ABS INT -z;
     "    ": RETURN
9999 SAVE "Wickel  DATA " LINE 1
```

Wenn alles eingegeben ist, erscheint ein Menu:

```
    WICKELDATENRECHNER

1) Netzspannung: 380 V
2) Polzahl: 2
3) Frequenz: 50 Hz
4) Durchmesser: 7 cm
5) Laenge: 7 cm
6) Induktion: 5600 Gauss
7) Nutenzahl: 24
8) Induktion: 4000-8000 Gauss
9) Leistungsschaetzung
0) Drahtstaerke ermitteln
OK j/n,aendern mit Zahlen 1-7
```

(Bei Eingabe von 1-7 kann der jeweilige Wert geändert werden.)

Bei der Eingabe von "j" wird die Drahtzahl pro Nute angezeigt.

Bei der Eingabe von "8" wird fortlaufend die Drahtzahl pro Nute bei Induktionen ab 4000 bis 8000, gestuft um jeweils 100, angezeigt. Die Funktion kann mit "a" (STOP) abgebrochen werden.

Bei Eingabe von "9" wird eine Konstante C angefordert und damit eine Leistungsschätzung ausgeführt. Nach dieser Leistung kann dann die Konstante nochmals genauer ermittelt und die Leistungsangabe damit korrigiert werden.

Bei Eingabe von "0" wird nach dem Nutquerschnitt (qmm) gefragt. Ist dieser nicht bekannt (Eingabe ENTER), dann die Daten einer ausgeführten Wicklung hier eingeben. Der Computer rechnet dann diese auf die neue Drahtzahl um.

Erfahrungsdaten

für die Konstante C:	für KW:
0.4 - 0.75	0.1 - 1
0.75- 1.5	1.1 - 7.5
1.5 - 2.4	7.6 - 25
2.4 - 3.2	26 - 100
3.2 - 4.3	101 -1000
4.3 - 5.5	über 1000

für Induktion	4500-8000 Gauß
schlechte Eisenqualität	4500-5000 Gauß
ältere Mot., mittl. "	5000-5500 Gauß
moderne Motoren	-8000 Gauß
Kleinmotoren 0,2-1 PS	4500-5000 Gauß
Motoren über 1000 PS	-9000 Gauß

Herbert Hartig, Postfach 323, 86803 Buchloe

Hubert Kracher
Kirchdorfer Str. 5
83064 Großholzhausen

Harald Lack
Heidenauer Str.5
83064 Raubling

Hallo Spectrum Freunde!!

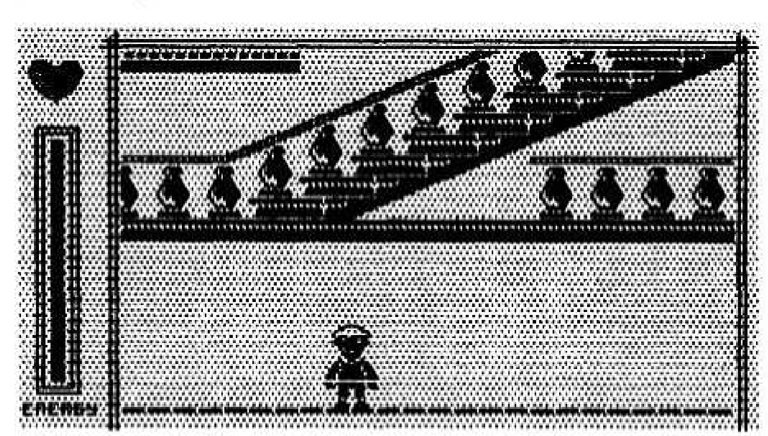
Nachdem wir uns in letzter Zeit sehr viel mit Adventurelösungen beschäftigt haben, widmen wir uns heute mal wieder einem Spiel aus der Ecke der Jump-'n'-Run Games. Es handelt sich um das Programm



das etwa gegen Ende 1992 auf den Markt kam. Es entstammt keinem der großen Software-Häuser, sondern wurde von einem Spectrum-User programmiert. Als sogenanntes Reader Game wurde es dann unter anderem von YOUR SINCLAIR veröffentlicht.

Da das zu erforschende Areal nicht sehr groß ist, es aber unserer Meinung nach nie schaden kann, wenn man sich bereits vorab ein wenig orientieren kann, haben wir beiliegenden Plan erstellt. Er hilft ein wenig dabei, sich etwas systematisch durch das Haus zu bewegen. Das eigentlich Schwierige an diesem Programm, ist die für die gestellte Aufgabe recht geringe Anzahl der Leben, die unsere Spielfigur zur Verfügung hat. Außerdem sind einige Dinge nur sehr schwer zu erreichen. Die Aufgabe, die wir zu lösen haben, besteht darin, alle im Haus verstreuten Schlüssel einzusammeln ohne vorher alle Leben auszuhauchen. Die an den Wänden befestigten kleinen Kreuze dienen übrigens der Energierückgewinnung für unsere Spielfigur. Damit wäre aber auch schon alles zum Spielablauf notwendige gesagt.

Abschließend noch ein paar kleine Tips, die vielleicht ganz nützlich sind, um das Spiel erfolgreich zu beenden:



- die Berührung der Schlüssel im Flug bremst unseren Fall, was unter Umständen ein Leben rettet

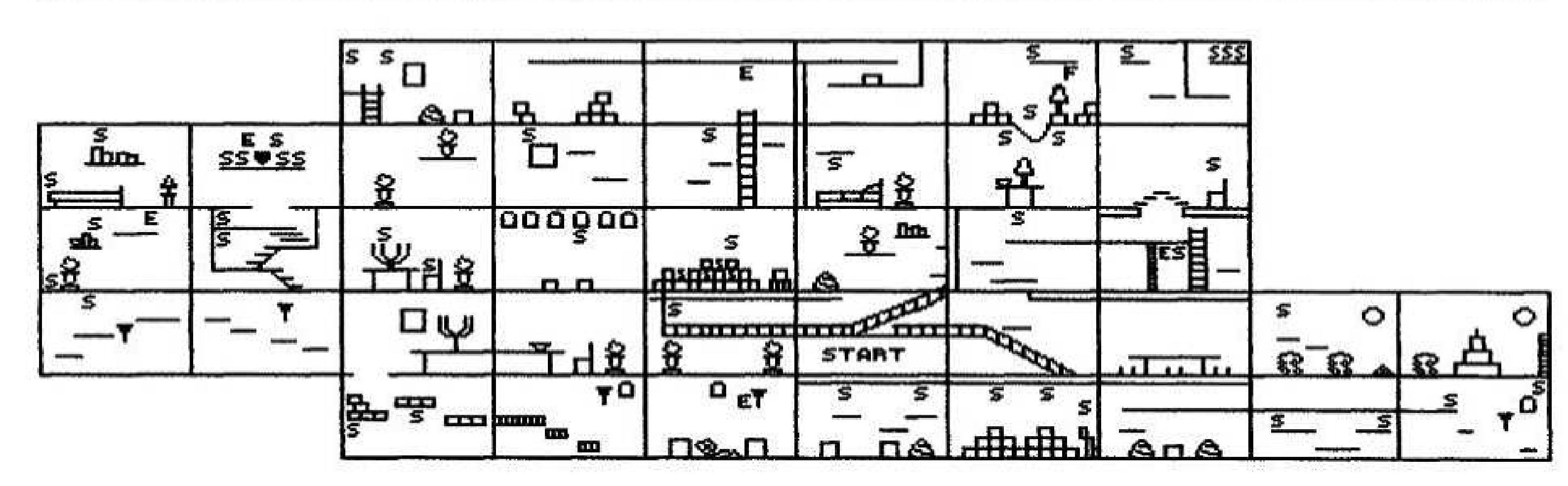
- den zweiten Raum von rechts im Dachgeschoß sollte man immer von links betreten, da man beim Nachgeben des Bodens dann auf der Lampe im Raum darunter landet und kein Leben verliert

- der Keller ist am schwierigsten. Man sollte ihn zuletzt erkunden

- man sollte, sobald man alle übrigen Schlüssel hat, in den letzten Kellerraum gehen um den entscheidenden Schlüssel einzusammeln

- es gibt insgesamt 48 Schlüssel

Soviel zu diesem Programm. Viel Spaß beim Nachspielen und Herumexperimentieren. Bis demnächst an dieser Stelle...





NEUE ZEICHENSATZE GANZ SCHNELL!

Wer kennt es nicht: Da hat man ein Demo gecoded, oder ein gutes Game geschrieben, nur der Standard Zeichensatz läßt alles so unprofessionell aussehen. Also sucht man einen Zeichensatz, findet die Diskette oder das Tape nicht und so weiter...

Damit ist jetzt Schluß!

Diese drei Einzeiler erzeugen einen kompletten Zeichensatz! Wen es interessiert wie das geht, der sollte einen Disassembler nehmen und das Listing betrachten. Es funktioniert kurz gesagt durch Manipulation des ROM-Zeichensatzes durch logische Verknüpfung beziehungsweise Rotation der Bytes.

Die Idee ist wiederum aus meiner 6 bändigen MC-Routinensammlung, nur weiß ich diesmal nicht, von wem sie stammt.

THICK FONT:
```
10 FOR I = 30000 TO 30029 : READ A: POKE I,A : NEXT I : RANDOMIZE USR 30000 : DATA 33,0,61,17,0,118,1,0,3,237,176,33,0,118,17,0,3,126,79,203,63,177,119,35,27,122,179,200,24,243
```

ITALIC FONT:
```
10 FOR I = 30000 TO 30041 : READ A: POKE I,A : NEXT I : RANDOMIZE USR 30000 : DATA 33,0,61,17,0,118,1,0,3,237,176,33,0,118,14,96,6,2,126,203,63,203,63,119,35,16,247,6,4,126,203,63,119,35,16,249,35,35,13,200,24,230
```

FUTURA FONT:
```
10 FOR I = 30000 TO 30035 : READ A: POKE I,A : NEXT I : RANDOMIZE USR 30000 : DATA 33,0,61,17,0,118,1,0,3,237,176,33,0,118,6,96,197,35,35,35,35,6,4,126,79,203,63,177,119,35,16,247,193,16,237,201
```

Wenn man den Code abspeichern möchte, so geht dies auch und zwar mit:

SAVE "FONT X" CODE 30208,768

Wer möchte und ein wenig Assembler versteht, kann versuchen, noch weitere Fonts durch probieren anderer Operationen am Systemzeichensatz zu generieren. Bis zur nächsten Folge des MC-Cauldrons ist dafür genug Zeit!

SpecTec, Christoph Kment, Trabertgasse 42, A-1130 Wien

Multiface-Pokes, Teil 11

Spiel	Poke(s)	Effekt
Rocky Horr.Pic.	48821,0	unendl.Leben
Rollaround	30900,0	
	+30941,0	unendl.Leben
Roller Coaster	38988,n	n<=255 Leben
Run For Gold	41079,0	Energie
	29866,60	extra Energ.
Rygar	51216,0	unendl.Zeit
	61577,0	unendl.Leben
SAS Combat Sim.	33039,0	unendl.Leben
	34013,0	Granaten
S.O.S.	33951,0	unendl.Leben
Saboteur	46998,0	unendl.Zeit
	29894,0	unendl.Leben
Saboteur 2	37122,0	unendl.Zeit
Sabre Wulf	44786,0	Keine Tiere
Sacred Armour	25007,99	
of Antiriad	+25008,255	Energie
Sai Combat	32421,n	n<=255 Leben
Salamander	38902,201	Unsichtbar
Samurai	41889,0	Kaufkraft
Samurai Warrior	33013,0	
	+37866,0	unendl.Leben
Sanxion	35028,0	Keine Aliens
Sceptre/Bagdad	59858,0	unendl.Leben
Scooby Doo	29614,0	unendl.Leben
Scuba Dive	55711,n	n<=255 Leben
Scumball	65036,0	unendl.Leben
Sentinel (The)	37490,0	unendl.Leben
Shockway Rider	46219,0	unendl.Leben
Sidearms	29411,127	unendl.Leben
Sidewize	36890,0	unendl.Leben

(ohne Gewähr, wird fortgesetzt)

Günther Marten Staulinie 12 26122 Oldenburg Tel. 0441 - 17976

* DTP LEICHT GEMACHT 31 * DOPPEL- UND MEHRFACHFONTS TEIL 11 *

Heute kommen wir zum vorerst letzten Teil der Doppel- und Mehrfachfont - Serie. Der vierteilige Font "BOOK 4" und der zweiteilige Font "EXPANDED 2" bilden somit das Abschlußlicht. "BOOK 4": Dieser DTP - Font arbeitet nur mit Großbuchstaben. Sonstige Zeichen wie Punkt, Komma, Plus, Minus etc. wurden nicht berücksichtigt. Bild 1 zeigt das Demo von "BOOK 4". Die Tastenbelegung:

t.book: l = ANFANG * A = ANSTATT EINES BUCHSTABEN * m = ZWISCHEN DEN BUCHSTABEN * r = ENDE

mt.book: Siehe t.book

m.book: Die Tasten A - Z sowie 0 - 9 sind mit der Spectrumtastatur identisch. Ä = SS + Y ** Ö = SS + D ** Ü = SS + U ** l = ANFANG ** SS + 2 = LEERKASTEN ** r = ENDE ** m = ZWISCHEN DEN BUCHSTABEN

b.book: Siehe t.book

BOOK 4

```
>sp 2
>F↓
B1MAMAMAMAMAMAMAr↓
B1MAMAMAMAMAMAMAr↓
C1MBMOMOMKM§M4Msr↓
B1MAMAMAMAMAMAMAr↓
>fill↓
>sp 3↓
>norm↓
```

m.book ABCDEFGHIJKL0123456780ÄÖÜ

t.book mt.book b.book

Der zweiteilige DTP - Font "EXPANDED 2": Dieser DTP - Font arbeitet nur mit Großbuchstaben. Sonstige Zeichen wie Punkt, Komma, Plus, Minus etc. wurden nicht berücksichtigt! Es folgt die Tastenbelegung:

t.expanded: Die Tasten A - Z sowie 0 - 9 sind mit der Spectrumtastatur identisch. Ä = SS + Y ** Ö = SS + D ** Ü = SS + U ** SS + 2 = LEERZEICHEN ** r = ENDE ** m = ZWISCHEN DEN BUCHSTABEN

b.expanded: In diesem Font werden die gleichen Zeichen verwendet wie im t.expanded-Font Siehe Bild 2 (rechts) als Beispiel.

EXPANDED 2

```
>sp 2
>large↓
E1EMXMPMAMNMDMEMDM§M2Mr↓
B1EMXMPMAMNMDMEMDM§M2Mr↓
>normal
>sp 3
>F↓
```

t.expanded

b.expanded

Im nächsten Teil geht es um das Programm "Kalender" Bis dahin wünsche ich noch viel Spaß mit DTP LG Teil 31 Günther Marten Oldenburg, den 22. 03. 94



Die heißen Juli- und Augusttage liegen hinter mir, mein QUICK-DOS-ROM hat seinen Geist aufgegeben (und hoffentlich schon wieder bekommen, wenn ihr diese Zeilen lesen könnt - Danke an Wilhelm), und, ach ja, das Quick-Move-Programm ist fertig.

Quick-Move 1.4

Jeder OPUS-Besitzer mit zwei Laufwerken (egal welcher Größe) und einem eingebauten RAM kann es benutzen, ganz egal, welches ROM eingebaut ist. Wie der Name schon vermuten läßt, wird der MOVE-Befehl schneller abgearbeitet, genauer, der MOVE-Befehl, welcher ein File von einer Disk auf eine andere Disk kopiert. Dies geht nun sieben mal so schnell wie zuvor!!!
Das Programm arbeitet mit allen Diskformaten, ja, es kann sogar SD-Disketten (178K, 40 Tracks) in einem DD-Laufwerk bearbeiten. Leseschwierigkeiten können durch wiederholtes Lesen teilweise behoben werden. Quell- und Zieldrive sind natürlich frei wählbar und diesmal ist das komplette Programm in MC geschrieben.
Zum Kopieren kann man eine Liste der Filenamen mit den Cursortasten bewegen (ein File oder 16 Files pro Tastendruck), die Liste alphabetisch sortieren lassen, durch einen Tastendruck alle wichtige Informationen eines Files (Art, Länge, etc.) abrufen, mit einem Tastendruck alle Files markieren, einzelne Files markieren, welche in der Reihenfolge des Markierens kopiert werden. Auf dem Bildschirm stehen Informationen über den Aufbau der Quell- und Zieldiskette, wie z.B. Anzahl der Files, freie Blöcke, benutzte Blöcke, Trackanzahl, Sektorengröße und vieles mehr. Ein Tastendruck startet das Kopieren. Existiert auf der Zieldisk schon ein File, welches denselben Namen hat, wie ein zu kopierendes File, kann man einen neuen Namen eingeben oder auch das File nicht kopieren. Alle auftretenden Fehler werden im Klartext auf dem Bildschirm ausgegeben, der Benutzer kann dann den Programmfortgang selbst bestimmen (Abbruch, File überspringen etc.).
Wie ihr das Programm bestellen könnt, steht hinten bei den Anzeigen.

Doch nun zum heutigen Inhalt der OPUS-Serie: Die Quick-Move-Disk hat einen Kopierschutz, weshalb jeder Besteller zwei Disketten mit dem Programm erhält. Dieser Kopierschutz startet das Programm automatisch, egal welcher Diskettenbefehl eingegeben wird (außer FORMAT). Wie geht dies?

Der OPUS-Info-Sektor

Wie in der letzten Folge erwähnt, benutzt das OPUS-System den ersten Sektor des ersten Tracks der Disk als Informationsquelle. Wird ein Diskettenbefehl (SAVE, LOAD, MERGE, VERIFY, MOVE, CAT, ERASE) ausgeführt, lädt die OPUS als erstes den Infosektor in ihr RAM und zwar an die Stellen 10112-10239, also an das Ende des RAMs. Es werden nur 128 Bytes geladen, da ja auch Disketten mit 128 Bytes pro Sektor formatiert werden können. Danach prüft die OPUS, ob der Sektor auch wirklich vorhanden ist (es könnte ja sein, daß kein RAM eingebaut ist) und ruft dann das MC-Programm auf, welches in diesem Sektor (und daher jetzt im OPUS-RAM) steht. Ist kein RAM eingebaut, werden die ROM-Diskdaten verwendet, dadurch ist in solchen Fällen nur das Formatieren von einseitigen Disks mit 40 Tracks zu 18 Sektoren der Größe 256 Bytes möglich.
Was das MC-Programm macht, geht aus folgendem Assemblerlisting hervor:

MC-Routine im OPUS-Info-Sektor

```
        JR    START

        DEFB  40          ;Trackanzahl
        DEFB  18          ;Sekt. pro Track
        DEFB  %01000000   ;Bitmap
```

Die Bitmap hat folgende Bedeutung:
Bits 7/6: Sektorgröße 00=128, 01=256, 10=512, 11=1024
Bit 5: wird aus alten Diskdaten im Speicher gelesen
Bit 4: 0=einseitig, 1=zweiseitig
Bit 0-3: werden aus alten Diskdaten im Speicher gelesen

```
        DEFB  19,5        ;keine Verwendung
```

Zu Beginn zeigt HL auf das erste Datenbyte, und IX auf die Diskdatentabelle (DDT) im OPUS-RAM

```
START   LD    A,(HL)      ;Trackzahl lesen
        LD    (IX+0),A    ;u. in DDT speichern
        INC   HL
        LD    A,(HL)      ;Sektorenzahl lesen
        LD    (IX+1),A    ;u. in DDT speichern
        LD    A,(IX+2)    ;alte Bitmap lesen
        AND   %00101111   ;Bits 0-3,5 behalten
        LD    D,A         ;zwischenspeichern
        INC   HL
        LD    A,(HL)      ;neue Bitmap lesen
```

```
AND %11010000 ;Bits 7,6,4 behalten
OR  D         ;verknüpfen mit alter
LD  (IX+2),A  ;bitmap und in DDT
RET           ;speichern, dann fertig
```

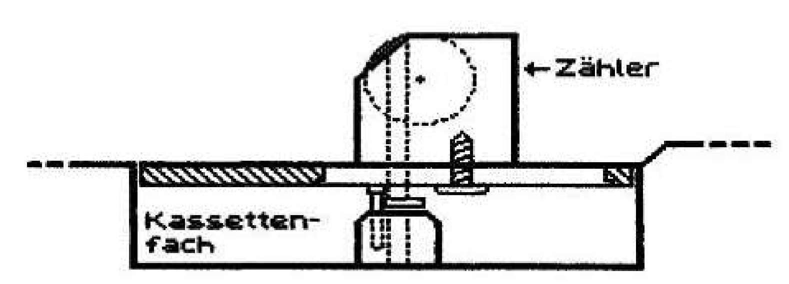
Jetzt wird vielleicht auch dem einen oder anderen schon klar, wie mein Kopierschutz funktioniert. Ich habe einfach das MC-Programm abgeändert, so daß sofort das Quick-Move-Programm geladen und gestartet wird.
Bei Aufruf des FORMAT-Befehls wird dieser Sektor natürlich nicht geladen. Dies hat zwei Gründe: Eine frische Disk (unformatiert) hat nämlich keinen solchen Sektor; würde der Sektor bei einer schon einmal formatierten Disk gelesen werden, könnte diese Disk nie in einem neuen Format formatiert werden, was ja auch nicht im Sinn der Sache sein kann.
Wie kommt der Info-Sektor auf die Disk?
Nachdem eine Disk formatiert wurde, wird einfach dieser Sektor, welcher im OPUS-ROM fest gespeichert ist, auf die Disk kopiert und anschließend noch der Catalogue geschrieben.
Wer das Programm genau betrachtet hat, stellt fest, daß keine Informationen über die Cataloguegröße gelesen wird. So kann eine neu formatierte Disk eine andere Cataloguegröße haben, als eine direkt zuvor benutzte Disk, obwohl sie mit deren Daten formatiert wurde.
Um die Cataloguelänge festzulegen, muß ein Wert in der Disk-Informationstabelle im OPUS-RAM geändert werden, worüber ich in einer der nächsten Folgen schreiben werde. Natürlich könnte man das MC-Programm des ersten Sektors erweitern, so daß auch die Cataloguelänge gelesen wird. Allerdings wird das so veränderte MC-Programm nicht automatisch auf die nächste formatierte Disk gespeichert, sondern muß "von Hand" daraufkopiert werden.
Veränderungen des MC-Programms könnten natürlich auch zur Erzeugung eines Virus auf der OPUS gut sein. Ganz so einfach ist die Sache aber nicht, da ja eine neue Disk sofort ihren Infosektor in das OPUS-RAM lädt und so dort das alte MC-Programm überschreibt. Wenn man allerdings die Tabellen in OPUS-RAM, welche zum Abarbeiten der SAVE-, LOAD-, etc. Befehle nötig sind, umschreibt, so daß bei deren Abarbeitung zuerst eine eigene Routine aufgerufen wird, welche ja in einem anderen Bereich des OPUS-RAMs sitzen kann, wäre es denkbar, daß...
Doch genug dieser bösen Gedanken, freuen wir uns doch, daß die OPUS Virenfrei ist!
Soviel für heute, Anregungen, Kritik, Wünsche können wie immer an mich herangetragen werden (oder ihr laßt es wie bisher immer, auch diesmal wieder sein). Ansonsten bleibt mir nichts, als euch viele fehlerfreie Disks zu wünschen, bis im Oktober dann (ihr könnt mich übrigens in Filderstadt treffen, wenn ihr kommt).

Helge Keller, Hermann-Löns-Weg 51
76307 Karlsbad, Tel.07202/6076

Ein Tip für Besitzer eines Spectrum +2 mit eingebautem Rekorder!

Nachteilig ist, daß er keinen Bandlängenzähler hat. Die Firma Conrad Electronic bietet als Restposten aus ehemaligen DDR-Beständen unter der Bestell-Nr. 490415-15 zu 0,60 DM Zähler an. Ich habe mir einen schicken lassen und eingebaut.

Da das eingebaute Laufwerk sehr flach gebaut ist, besteht keine Möglichkeit das Zählwerk wie vorgesehen über einen Riemen anzukoppeln. Ich habe daher das Zählwerk auf das Kassettenfach montiert. Dazu ritzt man sich den Mittelpunkt der aufspulenden, rechten Achse auf dem Klarsichtfenster des Kassettenfachs an und entfernt dieses aus dem Gerät. Vom Zähler zieht man die Seilscheibe ab und schneidet den Flansch an der Antriebswelle bis auf die Grundfläche ab.



Der Antriebswelle fehlt jetzt das Gegenlager. Dieses wird durch eine Bohrung, die genau dem Wellendurchmesser entspricht, im angeritzten Mittelpunkt des Klarsichtdeckels ersetzt. Die Welle des Zählers steckt man dann durch die Bohrung, zeichnet sich von unten die Bohrungen für die 3 Befestigungsschrauben an und führt diese aus. Mit 3 kleinen Blechschrauben wird der Zähler befestigt und so justiert, daß die Welle senkrecht steht und sich leicht drehen läßt.

Mit einem Seitenschneider knipst man die Welle bis auf etwa 3 mm ab. An diesen Wellenstumpf befestigt man einen kleinen Hebel. Damit dieser Hebel die Umdrehungen des Aufwickelkernes auf den Zähler überträgt, bohrt man mit einem 0,8 mm Bohrer (wie zum Platinenbohren) möglichst dicht an die Kante des Kernes ein oder zwei gegenüberliegende Löcher, in die abgeknipste Stecknadeln eingeklebt werden. Das Klarsichtfenster mit dem aufmontierten Zähler drückt man wieder in den Deckel.

Wenn jetzt eine Kassette eingelegt wird und das Fach geschlossen ist, übertragen die Stecknadeln die Umdrehungen vorwärts und rückwärts auf den Zähler. Der "Huckepackzähler" sieht etwas ungewohnt aus, aber funktioniert.

Günter Hartwig, Londonstr. 34
37079 Göttingen

The really rather boring Plus D column!!

Aus dem Diskmagazin AlchNews, mit freundlicher Genehmigung von Andy Davis (Editor) und Mark Harris (Autor). Ins deutsche gebracht vom WoMo-Team.

Wer den Artikel vom Vormonat gelesen hat wird sich vielleicht erinnern, daß sich mein +2A alle 15 Minuten oder so verabschiedet hat. Der Grund dafür war eine eine schlechte Verbindung des I/O Ports mit dem Plus D. Nun, das Problem ist behoben! Dies ist mein erster Artikel den ich komplett auf einem PC geschrieben habe. Ich verbrachte allerdings 12 Stunden damit, ein recht schlechtes Programm zu schreiben, das mir PC Files auf Plus D kopiert. Jetzt bin ich in der Lage, dies auf einem normalen Textprogramm mit spell check und allem anderen Mist zu schreiben.
Ihr werdet euch sicher wundern, warum ich dies erwähne? Nun, als ich das Programm schrieb lernte ich eine Menge über den Zugriff auf PC Disketten. Das klingt ein wenig ironisch? Gut, genug davon und weiter mit dem +D Stuff. Das letzte mal beschrieb ich Details für die Fehlerbehandlung und Lade-Routinen. Diesmal hoffe ich über die SAVE und DELETE Routinen zu berichten und beende dies mit dem Laden/Abspeichern von einzelnen oder mehreren Sektoren. Der Grund, warum ich das Laden von Sektoren nicht schon letztes mal beschrieb ist, das Laden und Saven im Prinzip das gleiche ist und ich nicht zweimal den gleichen Müll schreiben wollte, speziell nicht auf meinem "abstürzenden Spectrum".

SAVEN

Das erste, was ihr wissen solltet ist wie Files auf Diskette abgelegt werden samt extra Informationen. Am Anfang eines jeden Files, abgesehen von Snapshots, Open-Types etc., befindet sich ein 9 Byte Infoblock. Dieser ist ebenfalls im Directory abgelegt und gliedert sich wie folgt:

```
0  : Filetype (3 für Code/Screen$)
1-2: Länge
3-4: Startadresse
5-6: #FFFF (Code), Len-Vars (Basic)
7-8: Code Autostartadresse, Startzeile
```

Wenn ihr versucht, ein File ohne diese Informationen abzuspeichern, wärt ihr nicht mehr in der Lage, es aus Basic heraus zu laden. Das heißt aber, das es offensichtlich möglich ist, die Informationen wegzulassen wenn man es über den Z80 einlädt. Wenn dies so ist wäre es aber vorsorglich weise, das File ein Open-Type oder SPECIAL zu nennen.
Wenn ich jetzt mein Notebook finde werde ich die komplette Routine eintippen. Ihr glaubt doch wohl nicht, das ich den ganzen Mist im Kopf behalte?
Ok, der folgende Codeteil testet ob es sich um eine beschreibbare Diskette handelt und setzt die 9 Bytes an Informationen für das File. Wo sich die RET Anweisung befindet wäre es günstiger, ein JR in eine Errorroutine zu legen, die es euch erlaubt, weitere Versuche zu starten, aber das überlasse ich euch. Alles was ihr machen müßtet, wäre eine Meldung auszudrucken und die Routine neustarten zu lassen.

```
SAVE:   LD IX,#3AC3
        RST #8
        DB #40
        RET C (Drive empty)
        BIT 6,A
        RET NZ (Write protected)

        CALL ERROR (s. letztes Info)

        LD HL,FILENAME
        LD DE,#3E06
        LD BC,#A (10 Byte Filename)
        LDIR (Move Filename to +D)
        LD A,3
        LD (#3E10),A (Code File ID)
        INC A (A=4)
        LD (#3E05),A (+D Code File ID)
        LD HL,LENGTH
        LD (#3E11),HL (Länge!?)
        LD HL,START
        LD (#3E13),HL (noch Fragen?)
        LD HL,#FFFF
        LD (#3E15),HL
        INC HL
        LD (#3E17),HL
```

Der folgende Teil behandelt die ERASE Routine. Auch hier liegt es an euch, ob ihr eine Wahlmöglichkeit j/n einfügt. IHR TRAGT DIE KONSEQUENZEN! Beachtet, das der Filename ab Adresse #3E06 benötigt wird. Wildcards (*?) sind erlaubt, aber zum Saven nicht zu empfehlen!!!

```
ERASE:  LD A,#30
        CALL #9A5 (File suchen)
        JR NZ,READY (nicht gefunden)
TEST:   CALL #D93 (AUTO ERASE!
                  TODLICH!)
        LD (HL),0
        CALL #584
        JR TEST
```

Nun kommen wir zum Hauptteil. Als erstes öffnen wir ein neues File, dann rufen wir eine DOS Routine auf, um die Info an den Beginn des Files zu bringen. Dies arbeitet sowohl mit G+DOS als auch mit BetaDos, wenn ihr jedoch BetaDos mit einem erweiterten Katalog verwendet UND schon 80 Files belegt habt, wird die Routine mit dem Hinweis auf ein volles Directory abbrechen.

```
READY:    LD IX,#3AC3
          CALL #B1E (File öffnen)
          CALL #2879 (Move 9 Byte Header)
          LD HL,START
          LD DE,LENGTH
          CALL #850 (Save Bytes)
          CALL #B89 (File schließen)
          RET

ERROR:    IN A,(#E7) (s. letztes Info)
          LD HL,ERR_1
          EX (SP),HL
          LD (#2066),SP
          JP (HL)
ERR_1:    LD HL,0
          LD (#2066),HL
          OUT (#E7),A
          RET
```

Wie ihr seht, ist die wirkliche Save-Routine recht kurz und es sieht so aus, als wäre ich ganz einfach durchgekommen (Nicht?!), deshalb wende ich mich jetzt den Sektoren zu. Wenn ihr aber noch Fragen über irgendetwas soweit habt, dann schreibt mir an die untenstehende Adresse (in englisch). Ich kenne meinen 'Schreibstil' (wenn man das so nennen will).

SEKTOREN

Wie ich schon anfangs erwähnte, ist das Laden UND Saven von Sektoren das gleiche. Lediglich die Adressen haben sich geändert.

```
SECTORS:CALL PROTECTION
          LD A,DRIVE (1/2)
          LD D,TRACK (0-79,128-207)
          LD E,SECTOR (1-10)
          LD IX,ADDRESS
SAVE:     CALL #2FF2
LOAD:     CALL #2FD4
          RET

PROTECTION:
          LD IX,#3AC3
          RST #8
          DB #40
          RET C (Drive empty)
**************************************
          BIT 6,A (nur für SAVE_SEC)
          RET NZ (Write protected)
**************************************
          JP ERROR
```

Ich bin mir nicht sicher, ob dies auch mit BetaDos funktioniert, kann es mir aber vorstellen. Um auf der sicheren Seite zu sein, neige ich dazu, 'custom' Sektor-Routinen (z.B. lightening FAS...T) zu verwenden. Ruft zuerst den Protection Teil auf und versucht das folgende:

```
BOTH:     CALL PROTECTION
          LD HL,ADDRESS
          LD (STORE),HL
          LD D,TRACK
          LD E,SECTOR
          LD A,DRIVE
          CALL #705
          CALL ??_SEC
          RET

LD_SEC:   XOR A
          LD (#3DDB),A
LD_SC1:   LD IX,#3AC3
          CALL #667
          LD C,#80
          CALL #6F7
          LD HL,(STORE)*
          CALL #5E3
          CALL #60D
          JP LD_SC1

SV_SEC:   XOR A
          LD (#3DDB),A
SV_SC1:   LD IX,#3AC3
          CALL #667
          CALL #56C
          LD HL,(STORE)*
          CALL #599
          CALL #60D
          JR SV_SC1
STORE:    DW 0
```

Wenn ihr eine der beiden obigen Routinen ausprobiert, werden diese den Track/Sektor in DE an die Adresse, auf die HL zeigt, laden oder saven. Angenommen, ihr wollt mehrere Sektoren auf einmal laden, z.B die ersten 4 Tracks, dann probiert dieses Beispiel:

```
          CALL PROTECTION
          LD A,DRIVE
          CALL #705
          LD HL,#8000
          LD (STORE),HL
          LD DE,1
LOOP:     PUSH DE
          CALL LD_SEC
          LD HL,(STORE)
          INC H
          INC H
          LD (STORE),HL
          POP DE
          INC E
          LD A,E
```

```
CP #B
JR NZ,LOOP
LD E,1
INC D
LD A,D
CP 4
JR NZ,LOOP
RET
```

Diese Routine braucht ca. 2 Sekunden gegenüber 9 Sekunden, die die ROM Routine benötigt.
Zum Schluß (es ist immerhin 21.57 Uhr und ich möchte noch zu Abend essen) noch einige Kuriositäten. Alle davon sind nutzlos, aber ihr könnt sie auf Teufel komm raus ausprobieren:

```
LD IX,#3AC3
IN A,(#E7)
LD DE,1 (Track/Sektor)
CALL #667
LD C,FUNCTION
CALL #6F7
LD HL,BUFFER (versucht #80000)
CALL #5E3
OUT (#E7),A
EI (taking no chances)
RET
```

Für FUNCTION benutzt die folgenden Werte:

#90. Lade Track vom Sektor in E. Zum Beispiel: wenn E=1 wird der GANZE Track, wenn E=9 wird nur Track 9/10 geladen.

#E0. Load Format data for Track D. Absolut nutzlos!!!

#C0. Load CRC(?) data for Track. Bin mir auch nicht ganz sicher darüber. Auf jeden Fall müßt ihr es zweimal starten, wobei das Laufwerk vor dem zweiten Durchgang zum Stillstand gekommen sein muß. Auch dies ist absolut nutzlos!

Soviel für diesesmal. Bevor ich jedoch Schluß mache: Ist dort jemand der Erfahrung mit Ameisen hat? Ich weiß, das dies eine sonderbare Frage ist, aber ich bearbeite gerade meinen Garten und finde diese Tierchen überall. Das wäre nicht weiter schlimm, aber ich würde mich schuldig finden, wenn ich sie aus ihrem Lebensraum vertriebe. Verrückt, ich weiß, aber was soll ich tun? Ich bin wie ich bin.

Nächstesmal berichte ich wahrscheinlich über das Formatieren einzelner Tracks.
Wer weitere Aufklärungen oder Einzelheiten zum behandelten Thema braucht, oder wenn ihr irgendwelche Ideen habt, über was man in dieser Reihe noch berichten kann, kann mir an folgende Adresse schreiben.

Mark Harris, 1 Salters Lane
Faversham, Kent, ME13 8TD, England

Das BETA-Disk-IF

Dies ist der zweite Teil der neuen Serie für die Beta-Disk User, welcher in leicht anderer Form 1989 in den Infos des damaligen SUCW von Wilhelm Wiecke veröffentlicht wurde, nie jedoch beim SPC. Wir hoffen, allen Neu- und Wiedereinsteigern mit dieser Serie genügend Hintergrundinformation zu einem wirklich großartigen System zu geben.

Heute zum Betriebssystem der Version 5.03! Seit mehr als einem Jahr habe ich mich intensiv damit beschäftigt. Dazu mußte ich zuerst einmal alles disassemblieren. Das ergab einen Haufen bedrucktes Papier, so ca. 60 Seiten DIN A4 mit etwa 90 Zeilen pro Seite. Und da hab ich mich dann, wenn Zeit vorhanden war, daran gemacht und versucht, die ganze Sache so allmählich zu verstehen und erst einmal mit handschriftlichen Kommentaren zu versehen. Zuerst ging's sehr mühsam voran, aber so nach und nach findet man sich immer besser darin zurecht. So konnte ich bis heute zumindest einen Teil der Routinen entschlüsseln und am Rest wird weitergearbeitet! Nun habe ich vor, so nach und nach von allen Dingen allgemein brauchbare und interessante Routinen ausführlicher kommentiert als Assemblerlisting im Info zu beschreiben. Vielleicht besteht beim Einen oder Anderen Interesse an sowas!
Jetzt wird so mancher fragen, wie kommt man ohne Hilfsmittel wie Eprommer oder ähnliches überhaupt an das Betriebssystem heran? Ganz einfach geht's! Nur aus dem TRDOS heraus folgenden Befehl eingeben:

```
SAVE "Beta5.03" CODE 0,16384
```

und Sekunden später hat man den gesamten Eprom-Inhalt auf Diskette. Nun einen Disassembler laden und mit:

```
LOAD "Beta5.03" CODE xxxxx
```

den gespeicherten Code in einen geschützten Adressbereich laden und schon kann man alles auslesen.
Da ich natürlich sicher gehen wollte, daß das Betriebssystem auch wirklich vollständig und korrekt nach oben beschriebenem Verfahren wiedergegeben wurde, habe ich auch noch das ausgebaute Eprom ausgelesen und disassembliert! Und schon gab's eine Überraschung: Einzelne Codes stimmten nicht überein mit der vorher

gewonnenen Version und es ergaben sich unsinnige Befehlsfolgen! Die Ursache war schnell gefunden! Bei allen Zahlen, bei denen Bit 0 oder 7 gesetzt war, trat die Unstimmigkeit auf. Der Grund: vermutlich aus rein praktischen Gründen hat man die Datenleitungen für Bit 0 und 7 beim Eprom vertauscht, was ja belanglos ist, weil die Hardware dafür ausgelegt ist! Nur beim Auslesen in der üblichen Reihenfolge gibt's dann Unsinn. Zur Erklärung ein Beispiel:

```
                    Bit 7 6 5 4 3 2 1 0

                        0 0 0 0 0 0 0 1  =  Hex. 01  =  Dez.   1
Bits vertauscht         1 0 0 0 0 0 0 0  =  Hex. 80  =  Dez. 128

                        1 1 1 1 1 1 1 0  =  Hex. FE  =  Dez. 254
Bits vertauscht         0 1 1 1 1 1 1 1  =  Hex. 7F  =  Dez. 127
```

Nach dieser Erkenntnis läßt sich nun ein kleines Basicprogramm schreiben, was beim Auftreten der entsprechenden Zahlen korrigiert und mit den Werten der ersten Version vergleicht. Dazu müssen natürlich Beide gleichzeitig an unterschiedlichen Adressen im Speicher stehen (Programm Beispiel siehe unten)! Der Test ergab völlige Übereinstimmung beider Versionen.
Somit steht dem Disassemblieren der 1. Version nichts mehr im Wege. wer es nachvollziehen will, sollte möglichst einen Disassembler benutzen, der die Verwendung der Originalstartadresse 0 erlaubt, auch wenn der Code an beliebiger Stelle im Speicher steht. Das verbessert die Übersichtlichkeit des Listings. Günstig ist es außerdem, wenn man die Ausgabe der Spectrum-ROM spezifischen Datenbytes bei RST $08 bzw. RST $28 Befehlen abschalten kann, da diese im TRDOS nicht existieren (dafür jedoch ein RST $20, dazu demnächst mehr!).
Am Besten beginnt man damit, den gesamten Code als Hexdump mit gleichzeitiger ASCII-Zeichendarstellung durchzusehen. So lassen sich schnell Text-, Tabellen- und leere Eprom-Bereiche (enthalten $FF) finden. Die gefundenen Anfangs- und Endadressen bitte notieren! Nun kann man die übrigen Bereiche als Disassembler-Listing und die Tabellen usw. als Hex-Dump ausdrucken und erhält so ein zum größten Teil korrektes Listing! Diese Vorgehensweise empfiehlt sich immer beim Prüfen unbekannter MC-Programme.
Nun noch die Auflistung der Bereiche, die als Hex-Dump auszugeben sind (soweit mir bisher bekannt):

$0033-$0037	**leer (=$FF)**
$0360-$03AB	**Titeltext (* TR-DOS Ver 5.03 * usw.)**
$1000-$1017	**Text (Interface 1 fitted)**
$10A5-@1150	**Titeltext für LIST-Befehl**
$16F9-$170F	**Leer = ($FF)**
$1FB9-$1FC9	**Tabelle $01-$10 (Zweck noch unklar!)**
$2766-$283B	**Fehlermeld./Sonst. Texte**
$288C-$28D7	**Funktionsnummern mit jew. Einsprungadresse**
$29B3-$2A34	**Fehlermeldungstexte, CAT-Titeltext**
$2FF3-$3031	**Befehlscodes mit jeweil. Einsprungadresse**
$30FD-$31C7	**Befehlwort-Tabelle**
$31C8-$31F2	**Befehlscodes zu vorsteh. Tabelle**
$31FD-$3C00	**Leer (=$FF)**
$3C0C-$3CF9	**Leer (=$FF)**

Alles sonstige wird normal disassembliert. Einen Punkt muß man noch beachten: hinter einem $E7 => RST $20-Befehl folgen im TRDOS immer zwei Bytes, die eine Adresse für eine Routine im normalen Spectrum-Betriebssystem ergeben. Wie das funktioniert, werde ich vielleicht ein anderes Mal erklären. Für heute soll's erst mal reichen.

```
   1 REM ab 30000 => Eprom-Version
   2 REM ab 47000 => Disk-Version
   5 PRINT "Eprom";TAB 6;"Orig.";TAB
     11;"Korrig.";TAB 21;"Disk";TAB
     28;"Code"
  10 FOR n=30000 TO 46384
  20 LET x=n+17000
  30 GO SUB 100
  40 NEXT n
  50 STOP
 100 PRINT n;TAB 6;PEEK n;TAB 11;
 110 IF INT (PEEK n/2)=PEEK n/2 AND
     PEEK n>=128 THEN PRINT INVERSE
     1;PEEK n-127; INVERSE 0;: GO TO
     140
 120 IF INT (PEEK n/2)=PEEK n/2 AND
     PEEK n<=127 THEN PRINT INVERSE
     1;PEEK n+127; INVERSE 0;: GO TO
     140
 130 PRINT PEEK n;
 140 PRINT TAB 21;x;TAB 28;PEEK x
 150 RETURN
9989 STOP
9999 RANDOMIZE USR 15619: SAVE"
     Betatest"
```

Bis zum nächsten Mal!

*** Die Beta Connection ***
Jean Austermühle/Dirk Berghöfer

Das kurze Listing (zum Compilieren, z.B. HiSoft):

Der perfekte Kreis

```
   1 REM Alle Variablen sind INT
   2 REM Circle perfect by LCD
   3 REM Greetings to WoMoMel & Co
 100 LET x0=128: LET y0=87: LET r=87
 101 GO SUB 9900
 102 STOP
9900 LET x=0: LET da=INT (r/2)
9905 IF x>r THEN  RETURN
9910 IF da<0 THEN  LET da=da+r: LET r=
     r-1
9915 LET da=da-x-1
9920 LET z=y0+r: PLOT x0+x,z: PLOT x0-
     x,z: LET z=y0-r: PLOT x0+x,z:
     PLOT x0-x,z
9925 LET z=y0+x: PLOT x0+r,z: PLOT x0-
     r,z: LET z=y0-x: PLOT x0+r,z:
     PLOT x0-r,z
9930 LET x=x+1: GO TO 9905
```

L.C.D., Zieglergasse 98/9
A-1070 Wien, Österreich

ANTWORT

Zum Beitrag "Monitoranschluß am +2A" von Fred Dürkes:
Zur Schaltung möchte ich dringend empfehlen, eine Schutzdiode zwischen der Basis des Transistors und den COVO-Anschluß einzubringen, wie sie in den Geräten auch eingebaut ist (etwa 1N 4148, Kathode am Basisanschluß). Murphy lebt! Der TEA 2000 nur, solange die Kollektor-Basis-Strecke des Transistors gesund ist!

Heinz Schober, Taubenheimer Str. 18
01324 Dresden

FRAGEN

Wie kann man das RAM des +D-Interfaces auslesen? PEEK @,Adresse geht ja nicht.
Wie kommt man an Stecker für den AUX- und RS 232/MIDI-Anschluß des +2A?
Wer hat eine Programm-Anleitung für "LASER BASIC"?
Wer kann mir ein "SLOMO" (Programmablauf Verzögerungs Add-On mit Freezer-Taste), Hersteller: Nidd Valley Microproducts (1985) besorgen? Ggf. Schaltbild?

Heinz Schober, Taubenheimer Str. 18
01324 Dresden

ANZEIGEN

Highscreen-Farbmonitor (RGB, Scartanschluß) ca. 130,- DM; Plus D Interface mit Handbuch und Software ca. 110,- DM; 9 original verpackte 3-Zoll Disketten ca. 50,- DM; Spectrum +3 (eingebautes Laufwerk, externes Zweitlaufwerk, Druckerkabel, Handbuch und ca. 35 Arbeits-Disketten) ca. 290,- DM.
PC-Grafikkarte GM10-470, VLB, 1MB, integrierter AT-Bus Controller und Schnittstellen ca. 90,- DM.

Dieter Schulze-Kahleyß, Alb.-Schweitzer-Str. 21
71364 Winnenden, Tl. 07195/64404

Verkaufe Bücher: 1 ZX Microdrive-Buch von Andrew Pennel 5,- DM; 1 33 Programme für den Spectrum von R. Hülsmann 4,- DM; 1 Spaß und Profit von David Harwood 3,- DM; 1 Sinclair ZX Spectrum von Tim Hartnell 5,- DM.
Hardware: 1 Interface One (O.K.) für 18,- DM; 2 Interface One (defekt) für 3,50 DM.
Programm: Make a Chip (Inkognito Softw.) 5,- DM.

Günther Marten, Staulinie 12
26122 Oldenburg, Tel. 0441/17976 (n. 18.30 Uhr)

Für Historiker: Biete den BASIC-Lehrgang (6 Kassetten) von VEB Deutsche Schallplatten für 20,- DM an (unbenutzt).

Heinz Schober, Taubenheimer Str. 18
01324 Dresden

Es gab einmal eine Gruppe, die Musikstücke für die "Music Machine" geschrieben und gesammelt haben. Weiß jemand, wer das war, bzw. wer kann uns die Anschrift mitteilen? Oder wer von euch ist im Besitz von "Music Machine" Musikstücken?

Wolfgang Haller, Ernastraße 33
51069 Köln, Tel. 0221/685946

QUICK-MOVE 1.4: Kopiert einzelne Files von einer Disk auf eine andere (7 mal so schnell wie MOVE). Braucht OPUS mit zwei Laufwerken und RAM-Baustein, sowie Spectrum mit mind. 48 K. 2 Disketten (180K oder 720K, bitte angeben), 16-seitige Anleitung, Verpackung und Versand in Deutschland 15 DM, in Europa 20 DM (70FF, 8 £).

Helge Keller, Hermann-Löns-Weg 51
76307 Karlsbad, Tel.07202/6076

Wirklich schönes Stück: 48K SPECTRUM; ISSUE 2; normales Gehäuse, aber Plastiktasten; fast alle IC's gesockelt; ULA und Tastaturfolie neuwertig. Angebote bitte an:

Martin Hubbert, Hafenstr. 12, 45711 Datteln

SEIKOSHA GP-50 S; gut erhalten; 7 Farbbänder rot; 7 Rollen Papier; 100 DM + Porto bei

Peter Miosga, Holtbredde 11, 45711 Datteln